

Discovering Relevant Cross-Graph Cliques in Dynamic Networks

Loïc Cerf, Tran Bao Nhan Nguyen, and Jean-François Boulicaut

Université de Lyon, CNRS
INSA-Lyon, LIRIS, UMR5205, F-69621, France

Abstract. Several algorithms, namely CUBEMINER, TRIAS, and DATA-PEELER, have been recently proposed to mine closed patterns in ternary relations. We consider here the specific context where a ternary relation denotes the value of a graph adjacency matrix at different timestamps. Then, we discuss the constraint-based extraction of patterns in such dynamic graphs. We formalize the concept of δ -contiguous closed 3-clique and we discuss the availability of a complete algorithm for mining them. It is based on a specialization of the enumeration strategy implemented in DATA-PEELER. Indeed, clique relevancy can be specified by means of a conjunction of constraints which can be efficiently exploited. The added-value of our strategy is assessed on a real dataset about a public bicycle renting system. The raw data encode the relationships between the renting stations during one year. The extracted δ -contiguous closed 3-cliques are shown to be consistent with our domain knowledge on the considered city.

1 Introduction

Constraint-based mining is a popular framework for supporting pattern discovery. First, it provides more interesting patterns when the analyst specifies his/her subjective interestingness by means of a combination of primitive constraints. Then, this is known as a key issue to achieve efficiency and tractability: useful constraints can be deeply pushed into the extraction process such that it is possible to get complete (every pattern which satisfies the user-defined constraint is computed) though efficient algorithms.

In this paper, we focus on closed patterns that hold in ternary relations, i.e., Boolean cubes. They are a straightforward extension of the so-called formal concepts that hold in binary relations (see, e.g., [1]). This pattern domain has been extensively studied and efficient algorithms are available for mining constrained formal concepts. The extension of formal concept discovery to closed patterns in ternary relations has given rise to three proposals, namely CUBEMINER [2], TRIAS [3], and DATA-PEELER [4,5]. The main challenge of constraint-based closed pattern mining in 3-ary relations relies on the ability to push constraints during the extraction. To the best of our knowledge, only DATA-PEELER can mine closed patterns under a large class of constraints called *piecewise (anti)-monotonic constraints* [4,5]. This paper details how such constraints enable a

specialization of DATA-PEELER to the discovery of relevant patterns from dynamic graphs (e.g., dynamic interaction networks or dynamic co-interest graphs), considered as “stacks” of adjacency matrices, i.e., Boolean cubes.

Graph mining is a popular topic. Many researchers consider data analysis techniques for one large graph while others focus on graph pattern discovery from large collections of graphs. This article belongs to the latter family. More precisely, it deals with sets of timestamped graphs, i.e., dynamic networks. Our patterns are called δ -contiguous closed 3-cliques. Informally, they are maximal sets of vertices that are linked to each others and that run along some “almost” contiguous timestamped graphs. We show that this can be specified by means of primitive constraints which turn out to be piecewise (anti)-monotonic. As a result, the DATA-PEELER enumeration strategy can be specialized to support the search for knowledge nuggets from dynamic graphs, i.e., clique patterns that satisfy a wide range of user-defined constraints. Notice that such a formalization is new and has not been studied earlier. Due to the lack of space, we cannot recall the principles of the DATA-PEELER algorithm and we assume the reader can access to [4,5] for details.

In Sec. 2, we provide the problem setting where the mining task breaks into the satisfiability of a conjunction of constraints. They are proved piecewise (anti)-monotonic, i.e., they can be efficiently enforced by DATA-PEELER. Section 3 provides an experimental validation on a real dataset. Related work is discussed in Sec. 4, and Sec. 5 briefly concludes.

2 Extracting δ -Contiguous Closed 3-Cliques

2.1 Preliminaries

Let $\mathcal{T} \in \mathbb{R}^{|\mathcal{T}|}$ a finite set of timestamps. Let \mathcal{N} a set of nodes. A (possibly directed) graph is uniquely defined by its adjacency matrix $A \in \{0,1\}^{\mathcal{N} \times \mathcal{N}}$. A dynamic graph involving the nodes of \mathcal{N} along \mathcal{T} is uniquely defined by the $|\mathcal{T}|$ -tuple $(A_t)_{t \in \mathcal{T}}$ gathering the adjacency matrices of the graph at every timestamp $t \in \mathcal{T}$. Visually, such a stack of adjacency matrices can be seen as a $|\mathcal{T}| \times |\mathcal{N}| \times |\mathcal{N}|$ cube of 0/1 values. We write $a_{t,n^1,n^2} = 1$ (resp. $a_{t,n^1,n^2} = 0$) when, at the timestamp t , a link from n^1 to n^2 is present (resp. absent).

Example 1. Figure 1 depicts a dynamic directed graph involving four nodes a , b , c and d . Four snapshots of this graph are available at timestamps 0, 0.5, 2 and 3. Table 1 gives the related 4-tuple $(A_0, A_{0.5}, A_2, A_3)$.

Visually, a closed 3-set $(T, N^1, N^2) \in 2^{\mathcal{T}} \times 2^{\mathcal{N}} \times 2^{\mathcal{N}}$ appears as a combinatorial sub-cube of the data (modulo arbitrary permutations on any dimension) verifying both the *connection* and the *closedness* primitive constraints. Informally, it means that $T \times N^1 \times N^2$ only contains '1' values (connection), and any “super-cube” of (T, N^1, N^2) violates the connection constraint (closedness).

Definition 1 ($\mathcal{C}_{\text{connected}}$). *It is said that a 3-set (T, N^1, N^2) is connected, denoted $\mathcal{C}_{\text{connected}}((T, N^1, N^2))$, iff $\forall (t, n^1, n^2) \in T \times N^1 \times N^2, a_{t,n^1,n^2} = 1$.*

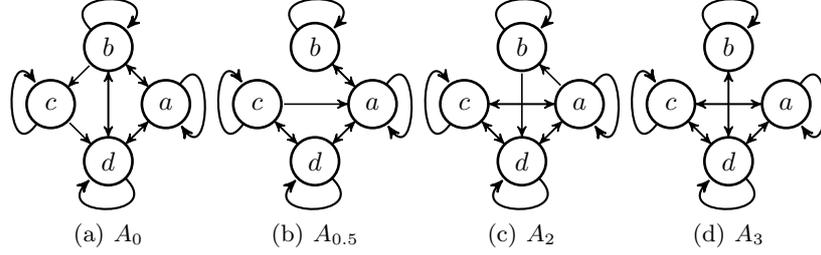


Fig. 1. Example of a dynamic directed graph ($\mathcal{N} = \{a, b, c, d\}$, $\mathcal{T} = \{0, 0.5, 2, 3\}$)

Table 1. ($A_0, A_{0.5}, A_2, A_3$) related to the dynamic graph depicted Fig. 1

	a	b	c	d	a	b	c	d	a	b	c	d	a	b	c	d
a	1	1	0	1	1	1	0	1	1	1	1	1	1	0	1	1
b	1	1	1	1	1	1	0	0	0	1	0	1	0	1	0	1
c	0	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1
d	1	1	0	1	1	0	1	1	1	0	1	1	1	1	1	1
	A_0				$A_{0.5}$				A_2				A_3			

Definition 2 ($\mathcal{C}_{\text{closed}}$). It is said that a 3-set (T, N^1, N^2) is closed, denoted

$$\mathcal{C}_{\text{closed}}((T, N^1, N^2)), \text{ iff } \begin{cases} \forall t \in \mathcal{T} \setminus T, \neg \mathcal{C}_{\text{connected}}(\{t\}, N^1, N^2) \\ \forall n^1 \in \mathcal{N} \setminus N^1, \neg \mathcal{C}_{\text{connected}}(T, \{n^1\}, N^2) \\ \forall n^2 \in \mathcal{N} \setminus N^2, \neg \mathcal{C}_{\text{connected}}(T, N^1, \{n^2\}) \end{cases} .$$

Definition 3 (Closed 3-set). It is said that (T, N^1, N^2) is a closed 3-set iff it satisfies the conjunction $\mathcal{C}_{\text{connected}}((T, N^1, N^2)) \wedge \mathcal{C}_{\text{closed}}((T, N^1, N^2))$.

Example 2. $(\{0, 2, 3\}, \{a, b, c, d\}, \{d\})$ is a closed 3-set in the toy dataset from Tab. 1: $\forall (t, n^1, n^2) \in \{0, 2, 3\} \times \{a, b, c, d\} \times \{d\}$, we have $a_{t, n^1, n^2} = 1$, and

$$\begin{cases} \forall t \in \{0.5\}, \neg \mathcal{C}_{\text{connected}}(\{t\}, \{a, b, c, d\}, \{d\}) \\ \forall n^1 \in \emptyset, \neg \mathcal{C}_{\text{connected}}(\{0, 2, 3\}, \{n^1\}, \{d\}) \\ \forall n^2 \in \{a, b, c\}, \neg \mathcal{C}_{\text{connected}}(\{0, 2, 3\}, \{a, b, c, d\}, \{n^2\}) \end{cases} .$$

$(\{2, 3\}, \{a, c, d\}, \{a, c, d\})$ and $(\{0, 3\}, \{b, d\}, \{b, d\})$ are two other closed 3-sets. $(\{0.5, 2, 3\}, \{c, d\}, \{c, d\})$ is not a closed 3-set because it violates $\mathcal{C}_{\text{closed}}$: its second set can be extended with a , i.e., $\mathcal{C}_{\text{connected}}(\{0.5, 2, 3\}, \{c, d\}, \{a\})$ holds.

2.2 Piecewise (Anti)-Monotonicity

DATA-PEELER can extract the complete collection of the closed 3-sets that hold in a dynamic graph. Its enumeration strategy is based on an enumeration tree which safely prunes the search space (i.e., no closed 3-set is missed) thanks to $\mathcal{C}_{\text{connected}}$, $\mathcal{C}_{\text{closed}}$ and any other user-defined piecewise (anti)-monotonic constraint. The efficient enforcement of any piecewise (anti)-monotonic constraint

\mathcal{C} is one of the main advantages of this principled algorithm. By “efficient”, we mean DATA-PEELER *sometimes* can, without any access to the data, affirm that the sub-tree rooted by the current enumeration node is empty of (not necessarily connected or closed) 3-sets satisfying \mathcal{C} . When the node is a leaf, it, not only *sometimes*, but *always* can check \mathcal{C} , hence ensuring the correctness (i.e., every extracted closed 3-set verifies \mathcal{C}). [5] provides a detailed presentation of the DATA-PEELER algorithm. Because of space constraints, this article only recalls the definition of piecewise (anti)-monotonicity.

Definition 4 (Monotonicity on the first argument). *A constraint \mathcal{C} is monotonic on the first argument iff $\forall (T, T', N^1, N^2) \in 2^{\mathcal{T}} \times 2^{\mathcal{T}} \times 2^{\mathcal{N}} \times 2^{\mathcal{N}}, T \subseteq T' \Rightarrow (\mathcal{C}(T, N^1, N^2) \Rightarrow \mathcal{C}(T', N^1, N^2))$. \mathcal{C} is anti-monotonic on the first argument iff $T \subseteq T' \Rightarrow (\mathcal{C}(T', N^1, N^2) \Rightarrow \mathcal{C}(T, N^1, N^2))$.*

In a similar way, the monotonicity (or anti-monotonicity) on the second (or third) argument can be defined.

Example 3. Consider the following constraint:

$$\text{A 3-set } (T, N^1, N^2) \text{ is 8-large} \Leftrightarrow |T \times N^1 \times N^2| \geq 8 .$$

This constraint is monotonic on the first argument. Indeed, $\forall (T, T', N^1, N^2) \in 2^{\mathcal{T}} \times 2^{\mathcal{T}} \times 2^{\mathcal{N}} \times 2^{\mathcal{N}}, T \subseteq T' \Rightarrow (|T \times N^1 \times N^2| \geq 8 \Rightarrow |T' \times N^1 \times N^2| \geq 8)$. It is monotonic on the second and on the third argument too.

The class of piecewise (anti)-monotonic constraints not only contains every constraint which is either monotonic or anti-monotonic on each of its arguments but also every constraint whose expression is such that, when giving a different variable to every occurrence of the three original arguments (related to the three sets \mathcal{T} , \mathcal{N} and \mathcal{N}), the newly obtained constraint is either monotonic or anti-monotonic on each of its arguments.

Example 4. Assume $\mathcal{T} \in \mathbb{R}_+^{|\mathcal{T}|}$ and consider the following constraint:

$$\mathcal{C}_{16\text{-small-in-average}}((T, N^1, N^2)) \Leftrightarrow T \neq \emptyset \wedge \frac{\sum_{t \in T} t}{|T|} \leq 16 .$$

This constraint is both monotonic and anti-monotonic on the second and the third argument (neither N^1 nor N^2 appearing in the expression of the constraint) but it is neither monotonic nor anti-monotonic on the first argument. However, giving three different variables T_1 , T_2 and T_3 to each of the occurrences of T creates this new constraint which is monotonic on the first and third arguments (T_1 and T_3) and anti-monotonic on the second one (T_2):

$$\mathcal{C}'_{16\text{-small-in-average}}(T_1, T_2, T_3, N^1, N^2) \equiv T_1 \neq \emptyset \wedge \frac{\sum_{t \in T_2} t}{|T_3|} \leq 16 .$$

Therefore $\mathcal{C}_{16\text{-small-in-average}}$ is, by definition, piecewise (anti)-monotonic.

Obviously, a conjunction of piecewise (anti)-monotonic constraints is piecewise (anti)-monotonic. We now formalize the search for relevant patterns (the, so-called, δ -contiguous closed 3-cliques) as the correct and complete extraction of closed 3-sets satisfying a conjunction of piecewise (anti)-monotonic constraints.

2.3 δ -Contiguous Closed 3-Sets

Given $\delta \in \mathbb{R}_+$, a δ -contiguous 3-set is such that it is possible to browse the whole subset of timestamps by jumps from one timestamp to another without exceeding a delay of δ for each of these jumps:

Definition 5 (δ -contiguity). *It is said that a 3-set (T, N^1, N^2) is δ -contiguous, denoted $\mathcal{C}_{\delta\text{-contiguous}}((T, N^1, N^2))$, iff $\forall t \in [\min(T), \max(T)], \exists t' \in T$ s.t. $|t - t'| \leq \delta$.*

The constraint $\mathcal{C}_{\delta\text{-contiguous}}$ is piecewise (anti)-monotonic.

Proof. Let $\mathcal{C}'_{\delta\text{-contiguous}}$ the following constraint:

$$\begin{aligned} & \mathcal{C}'_{\delta\text{-contiguous}}(T_1, T_2, T_3, N_1, N_2) \\ \equiv & \forall t \in [\min(T_1), \max(T_2)], \exists t' \in T_3 \text{ s.t. } |t - t'| \leq \delta . \end{aligned}$$

The three arguments T_1, T_2 and T_3 substitute the three occurrences of T (in the definition of $\mathcal{C}_{\delta\text{-contiguous}}$). $\mathcal{C}'_{\delta\text{-contiguous}}$ is monotonic in on its third argument and anti-monotonic on its first and second arguments ($T \subseteq T_1 \Rightarrow \min(T) \geq \min(T_1)$ and $T \subseteq T_2 \Rightarrow \max(T) \leq \max(T_2)$). Moreover, since its two last arguments do not appear in its expression, $\mathcal{C}'_{\delta\text{-contiguous}}$ is both monotonic and anti-monotonic on them. Therefore, by definition, $\mathcal{C}_{\delta\text{-contiguous}}$ is piecewise (anti)-monotonic. \square

$\mathcal{C}_{\text{connected}} \wedge \mathcal{C}_{\delta\text{-contiguous}}$ being stronger than $\mathcal{C}_{\text{connected}}$ alone, a related and weaker closedness constraint can be defined. Intuitively, a δ -closed 3-set is closed w.r.t. both \mathcal{N} sets and to the timestamps of \mathcal{T} in the vicinity of those inside the 3-set. Hence, a timestamp that is too far away (delay exceeding δ) from any timestamp inside the 3-set, cannot prevent its δ -closedness.

Definition 6 (δ -closedness). *It is said that a 3-set (T, N^1, N^2) is δ -closed, denoted $\mathcal{C}_{\delta\text{-closed}}((T, N^1, N^2))$, iff*

$$\begin{cases} \forall t \in \mathcal{T} \setminus T, (\exists t' \in T \text{ s.t. } |t - t'| \leq \delta \Rightarrow \neg \mathcal{C}_{\text{connected}}(\{\{t\}, N^1, N^2\})) \\ \forall n^1 \in \mathcal{N} \setminus N^1, \neg \mathcal{C}_{\text{connected}}((T, \{n^1\}, N^2)) \\ \forall n^2 \in \mathcal{N} \setminus N^2, \neg \mathcal{C}_{\text{connected}}((T, N^1, \{n^2\})) \end{cases} .$$

The constraint $\mathcal{C}_{\delta\text{-closed}}$ is piecewise (anti)-monotonic.

Proof. Let $\mathcal{C}'_{\delta\text{-closed}}$ the following constraint:

$$\begin{aligned} & \mathcal{C}'_{\delta\text{-closed}}(T_1, T_2, T_3, T_4, N_1^1, N_2^1, N_3^1, N_1^2, N_2^2, N_3^2) \\ \equiv & \begin{cases} \forall t \in \mathcal{T} \setminus T_1, (\exists t' \in T_2 \text{ s.t. } |t - t'| \leq \delta \Rightarrow \neg \mathcal{C}_{\text{connected}}(\{\{t\}, N_1^1, N_1^2\})) \\ \forall n^1 \in \mathcal{N} \setminus N_2^1, \neg \mathcal{C}_{\text{connected}}((T_3, \{n^1\}, N_2^2)) \\ \forall n^2 \in \mathcal{N} \setminus N_3^2, \neg \mathcal{C}_{\text{connected}}((T_4, N_3^1, \{n^2\})) \end{cases} . \end{aligned}$$

$\mathcal{C}'_{\delta\text{-closed}}$ is anti-monotonic on its second argument and monotonic on all its other arguments. Therefore, by definition, $\mathcal{C}_{\delta\text{-closed}}$ is piecewise (anti)-monotonic. \square

Definition 7 (δ -contiguous closed 3-set). *It is said that (T, N^1, N^2) is a δ -contiguous closed 3-set iff it satisfies $\mathcal{C}_{\text{connected}} \wedge \mathcal{C}_{\delta\text{-contiguous}} \wedge \mathcal{C}_{\delta\text{-closed}}$.*

Example 5. $(\{2, 3\}, \{a, b, c, d\}, \{d\})$ is a 1.75-contiguous closed 3-set in the toy dataset from Tab. 1. However, it is neither 0.5-contiguous (the timestamps 2 and 3 are not close enough) nor 2-closed (0 can extend the set of timestamps). This illustrates the fact that the number of δ -contiguous closed 3-sets is not monotonic in δ .

A δ -contiguous closed 3-set is an obvious generalization of a closed 3-set. Indeed, $\forall \delta \geq \max(\mathcal{T}) - \min(\mathcal{T}), \mathcal{C}_{\delta\text{-contiguous}} \equiv \text{true}$ and $\mathcal{C}_{\delta\text{-closed}} \equiv \mathcal{C}_{\text{closed}}$.

2.4 δ -Contiguous Closed 3-Cliques

We want to extract sets of nodes that are entirely interconnected. In this context, a 3-set (T, N^1, N^2) where $N^1 \neq N^2$ is irrelevant.

Definition 8 (Symmetry). *It is said that (T, N^1, N^2) is symmetric, denoted $\mathcal{C}_{\text{symmetric}}((T, N^1, N^2))$, iff $N^1 = N^2$.*

$\mathcal{C}_{\text{symmetric}}((T, N^1, N^2)) \equiv N^1 \subseteq N^2 \wedge N^2 \subseteq N^1$ is an equivalent definition to the symmetry constraint. In this form, a piecewise (anti)-monotonic constraint is identified.

Proof. Let $\mathcal{C}'_{\text{symmetric}}$ the following constraint:

$$\mathcal{C}'_{\text{symmetric}}(T, N_1^1, N_2^1, N_1^2, N_2^2) \equiv N_1^1 \subseteq N_1^2 \wedge N_2^2 \subseteq N_2^1 .$$

$\mathcal{C}'_{\text{symmetric}}$ is monotonic on its third and fourth arguments (N_2^1 and N_1^2) and anti-monotonic on its second and fifth arguments (N_1^1 and N_2^2). Moreover, since the first argument (T) does not appear in the expression of $\mathcal{C}'_{\text{symmetric}}$, this constraint is both monotonic and anti-monotonic on this argument. Therefore, by definition, $\mathcal{C}_{\text{symmetric}}$ is piecewise (anti)-monotonic. \square

$\mathcal{C}_{\text{connected}} \wedge \mathcal{C}_{\delta\text{-contiguous}} \wedge \mathcal{C}_{\text{symmetric}}$ being stronger than $\mathcal{C}_{\text{connected}} \wedge \mathcal{C}_{\delta\text{-contiguous}}$, a related and weaker closedness constraint can be defined. Intuitively, if not *both* the line and the column pertaining to a node n can *simultaneously* extend a 3-set without breaking $\mathcal{C}_{\text{connected}}$, the closedness is not violated:

Definition 9 (Symmetric δ -closedness). *It is said that a 3-set (T, N^1, N^2) is symmetric δ -closed, denoted $\mathcal{C}_{\text{sym-}\delta\text{-closed}}((T, N^1, N^2))$, iff*

$$\begin{cases} \forall t \in \mathcal{T} \setminus T, (\exists t' \in T \text{ s.t. } |t - t'| \leq \delta \Rightarrow \neg \mathcal{C}_{\text{connected}}(\{\{t\}, N^1, N^2\})) \\ \forall n \in \mathcal{N} \setminus (N^1 \cap N^2), \neg \mathcal{C}_{\text{connected}}((T, N^1 \cup \{n\}, N^2 \cup \{n\})) \end{cases} .$$

The constraint $\mathcal{C}_{\text{sym-}\delta\text{-closed}}$ is piecewise (anti)-monotonic.

Proof. Let $\mathcal{C}'_{\text{sym-}\delta\text{-closed}}$ the following constraint:

$$\begin{aligned} & \mathcal{C}'_{\text{sym-}\delta\text{-closed}}((T_1, T_2, T_3, N_1^1, N_2^1, N_3^1, N_1^2, N_2^2, N_3^2)) \\ \equiv & \begin{cases} \forall t \in \mathcal{T} \setminus T_1, (\exists t' \in T_2 \text{ s.t. } |t - t'| \leq \delta \Rightarrow \neg \mathcal{C}_{\text{connected}}(\{\{t\}, N_1^1, N_1^2\})) \\ \forall n \in \mathcal{N} \setminus (N_2^1 \cap N_2^2), \neg \mathcal{C}_{\text{connected}}((T, N_3^1 \cup \{n\}, N_3^2 \cup \{n\})) \end{cases} . \end{aligned}$$

$\mathcal{C}'_{\text{sym-}\delta\text{-closed}}$ is anti-monotonic on its second argument (T_2) and monotonic on all its other arguments. Therefore, by definition, $\mathcal{C}_{\text{sym-}\delta\text{-closed}}$ is piecewise (anti)-monotonic. \square

Definition 10 (δ -contiguous closed 3-clique). *It is said that (T, N^1, N^2) is a δ -contiguous closed 3-clique iff it satisfies $\mathcal{C}_{\text{connected}} \wedge \mathcal{C}_{\delta\text{-contiguous}} \wedge \mathcal{C}_{\text{symmetric}} \wedge \mathcal{C}_{\text{sym-}\delta\text{-closed}}$.*

Example 6. Two out of the three closed 3-sets illustrating Ex. 2 are symmetric: $(\{2, 3\}, \{a, c, d\}, \{a, c, d\})$ and $(\{0, 3\}, \{b, d\}, \{b, d\})$. $(\{0.5, 2, 3\}, \{c, d\}, \{c, d\})$ is not closed w.r.t. $\mathcal{C}_{\text{closed}}$ (see Ex. 2) but it is symmetric 1.75-closed. Indeed, the node a cannot simultaneously extend its second and third sets of elements without violating $\mathcal{C}_{\text{connected}}$.

Since $\mathcal{C}_{\text{connected}}$, $\mathcal{C}_{\delta\text{-contiguous}}$, $\mathcal{C}_{\text{symmetric}}$ and $\mathcal{C}_{\text{sym-}\delta\text{-closed}}$ are piecewise (anti)-monotonic, DATA-PEELER can handle them all, i.e., it can efficiently compute every 3-set satisfying $\mathcal{C}_{\text{connected}} \wedge \mathcal{C}_{\delta\text{-contiguous}} \wedge \mathcal{C}_{\text{symmetric}} \wedge \mathcal{C}_{\text{sym-}\delta\text{-closed}}$. In practical settings, the complete collection of the δ -contiguous closed 3-cliques is huge. It makes sense to constrain further the extraction tasks (i.e., to enforce also a new user-defined piecewise (anti)-monotonic constraint \mathcal{C}) to take subjective interestingness into account and to focus on more relevant patterns.

3 Experimental Results

Vélo'v is a bicycle rental service run by the city of Lyon, France. 338 Vélov stations are spread over this city. At any of these stations, the users can take a bicycle and return it to any other station. Whenever a bicycle is rented or returned, this event is logged. We focus here on the data generated along the year 2006. These data are aggregated to obtain one graph per period of time (we chose a period of 30 minutes). The set of nodes \mathcal{N} of such a graph corresponds to the Vélov stations. Its edges are labelled with the total number of rides in 2006 between the two linked stations (whatever their orientation) during the considered period of time. Setting a threshold allows to select the most significant edges. Many sensible strategies can be chosen to fix this threshold (which can be different between the graphs). Here is the strategy we opted for:

Binarization. Given a graph whose edges are labelled by real values, let m be the maximum of these values. The threshold is fixed to $0.2 \times m$.

Once the thresholds set, all edges linked to some station may be considered insignificant. Such an infrequently used station is removed from the dynamic graph. In our experiments, 204 stations remained after applying the binarization.

On an AMD SempronTM 2600+ running a GNU/LinuxTM operating system, our implementation in C++ extracts all 3849 closed 3-cliques within about three minutes and a half (extracting all 122581 closed 3-sets takes over nine minutes). Nevertheless, to assess, by hand, the quality of the extracted δ -contiguous closed 3-cliques, the returned collection must be small. Hence stronger constraints are enforced. The minimal number of Vélov stations that must be involved in a δ -contiguous closed 3-clique is set to 6 and the minimal number of periods to 4. Thanks to these constraints, only three 0.5-continuous closed 3-cliques are

returned within one minute and 49 seconds. Two of them take place during the evening (they start at half past 19) and gather stations that are in the center of Lyon. They differ by one station and one of them runs along one more time period. The third 0.5-contiguous closed 3-clique is displayed in Fig 2(a). The circles stand for the geographical positions of the Vélov stations. The filled ones are involved in the shown pattern. The disposition of the stations follows one of the greatest street in Lyon: “Cours Gambetta”. Obviously it is much used by the riders during the evening. The outlying Vélov station is, overall, the most frequently used one: “Part-Dieu/Vivier-Merle”. At this place, the rider finds the unique commercial center in Lyon, the main train station, etc. Extracting, with the same minimal size constraints, the 1-contiguous closed 3-cliques provides a collection of nine patterns. Among them, the three 0.5-contiguous closed 3-cliques are found unaltered; some slight variations of them are found (one or two stations are changed); one pattern takes place during the morning (to obtain patterns involving night periods the constraints must be weakened a lot: nightly rides do not comply much with a model). The majority of the extracted 1-contiguous closed 3-cliques involves Vélov stations in the city center. Figure 2(b) depicts one of them. The disposition of the stations follows the street binding the two most active districts in Lyon: “Rue de la Part-Dieu”. The outlying Vélov station is, overall, one of the most frequently used: “Opéra”. At this place, the rider can find, not only the opera, but also the town hall, a cinema, bars, etc.

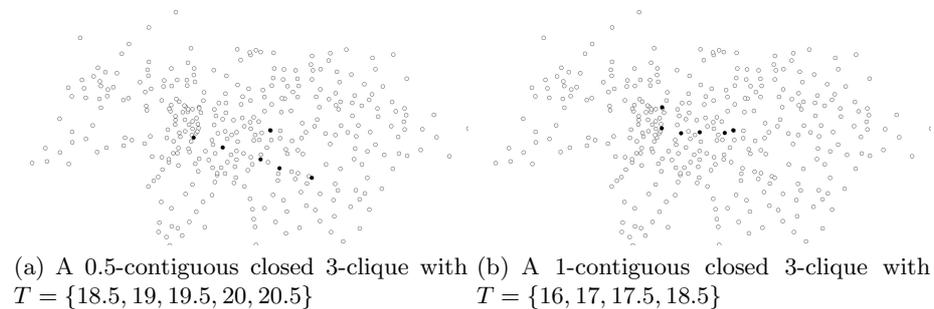


Fig. 2. Two closed 3-cliques extracted under strong constraints

4 Related Work

DATA-PEELER faces two competitors able to extract all closed 3-sets from ternary relations: CUBEMINER [2] and TRIAS [3]. None of them have the generality of DATA-PEELER. In particular, they cannot deal with n -ary relations and cannot enforce any piecewise (anti)-monotonic constraints. This latter remark makes them harder to specialize in the extraction of δ -contiguous closed 3-cliques. Furthermore, [5] shows that DATA-PEELER outperforms both of them by orders of magnitude. The interested reader will refer to the “Related Work” section of this article for a detailed analysis of what makes DATA-PEELER more efficient.

Collections of large graphs were built to help in understanding genetics. These graphs commonly have tens of thousands of nodes and are much noisy. There is a need to extract patterns crossing such graphs, e.g., patterns that remain valid across several co-expression graphs or patterns crossing the data pertaining to physical interactions between molecules (e.g., protein-gene) and more conceptual data (e.g., co-expression of genes). One of the most promising pattern helping in these tasks is the closed 3-clique or, better, the closed quasi-3-clique. CLAN [6] is able to extract closed 3-cliques from collections of large and dense graphs. Crochet+ [7], Cocain* [8] and Quick [9] are the state-of-the-art extractors of closed quasi-3-cliques. They all use the same definition of noise tolerance: every node implied in a pattern must have, in every graph independently from the others, a degree exceeding a user-defined proportion of the maximal degree it would reach if the clique was exact. An ongoing research for us is to generalize $\mathcal{C}_{\text{connected}}$ and $\mathcal{C}_{\text{closed}}$ to allow some fault-tolerance, i.e., to extract closed quasi-3-sets. Unlike the previous approaches, such a generalization of DATA-PEELER would allow the discovery of (possibly δ -contiguous) closed quasi-3-cliques in dynamic *directed* graphs. Discussing this further is out of the scope of this paper.

The δ -contiguity stems from an analogous constraint, called *max-gap* constraint, initially applied to sequence mining. It was introduced in the GSP approach [10]. The *min-gap* and the *window size* constraints [10] uses as well could be enforced in our approach too. Nevertheless, in [10], these constraints modify the enumeration order, whereas, in our approach, they reduce the search space and let the enumeration strategy unaltered. Furthermore, in the context of [10], the considered datasets are multiple sequences of itemsets and the extracted patterns are sub-sequences of itemsets whose order (but not position in time) is to be respected in all (1-dimensional) supporting sequences, whereas, in our approach, the supporting domain contains (2-dimensional) graphs that must be aligned in time. Notice that the max-gap constraint was used in other contexts. For example, [11] enforces it to extract frequent episodes and [12] extracts, under a max-gap constraint, frequent sub-sequences whose support is the sum of the number of repetitions in all sequences of the dataset.

5 Conclusion

This article deals with mining δ -contiguous closed 3-cliques. The constraints imposed to achieve this goal were proved piecewise (anti)-monotonic such that DATA-PEELER efficiently handles them all. Notice that ad-hoc optimizations for the studied conjunction of constraints have been already studied and implemented. However, they need the presentation of many technical details. Due to the severe space constraint, we decided to emphasize how, in its original form, the DATA-PEELER closed n -set extractor can be specialized in computing all δ -contiguous closed 3-cliques. We mentioned that an ongoing work concerns the declarative specification of quasi-cliques to support the discovery of more relevant patterns from real data sets. Notice also that DATA-PEELER can mine closed n -sets (or cliques) with n an arbitrary integer greater or equal to 2. It

can also enforce the contiguity of the patterns on several dimensions at the same time (possibly with different δ values). More generally, DATA-PEELER can mine closed n -sets adapted to any specific problem that can be expressed in terms of piecewise (anti)-monotonic constraints what appears definitively useful for many dynamic graph analysis processes.

Acknowledgments. This work has been funded by ANR BINGO2 (MDCO 2007-2010). Tran Bao Nhan Nguyen has contributed to this study thanks to a Research Attachment programme between the Nanyang Technological University (Singapore), where he is an undergraduate student, and INSA-Lyon. Finally, we thank Dr. J. Besson for exciting discussions.

References

1. Ganter, B., Stumme, G., Wille, R.: Formal Concept Analysis, Foundations and Applications. Springer, Heidelberg (2005)
2. Ji, L., Tan, K.-L., Tung, A.K.H.: Mining frequent closed cubes in 3D data sets. In: VLDB 2006: Proc. of the 32nd Int. Conf. on Very Large Data Bases, pp. 811–822. VLDB Endowment (2006)
3. Jäschke, R., Hotho, A., Schmitz, C., Ganter, B., Stumme, G.: TRIAS—an algorithm for mining iceberg tri-lattices. In: ICDM 2006: Proc. of the Sixth Int. Conf. on Data Mining, pp. 907–911. IEEE Computer Society, Los Alamitos (2006)
4. Cerf, L., Besson, J., Robardet, C., Boulicaut, J.F.: Data-Peeler: Constraint-based closed pattern mining in n -ary relations. In: SDM 2008: Proc. of the Eighth SIAM Int. Conf. on Data Mining, pp. 37–48. SIAM, Philadelphia (2008)
5. Cerf, L., Besson, J., Robardet, C., Boulicaut, J.F.: Closed patterns meet n -ary relations. ACM Trans. on Knowledge Discovery from Data 3(1) (March 2009)
6. Wang, J., Zeng, Z., Zhou, L.: CLAN: An algorithm for mining closed cliques from large dense graph databases. In: ICDE 2006: Proc. of the 22nd Int. Conf. on Data Engineering, pp. 73–82. IEEE Computer Society, Los Alamitos (2006)
7. Jiang, D., Pei, J.: Mining frequent cross-graph quasi-cliques. ACM Trans. on Knowledge Discovery from Data 2(4) (January 2009)
8. Zeng, Z., Wang, J., Zhou, L., Karypis, G.: Out-of-core coherent closed quasi-clique mining from large dense graph databases. ACM Trans. on Database Systems 32(2), 13–42 (2007)
9. Liu, G., Wong, L.: Effective pruning techniques for mining quasi-cliques. In: ECML PKDD 2008: Proc. of the European Conf. on Machine Learning and Knowledge Discovery in Databases - Part II, pp. 33–49. Springer, Heidelberg (2008)
10. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalizations and performance improvements. In: Apers, P.M.G., Bouzeghoub, M., Gardarin, G. (eds.) EDBT 1996. LNCS, vol. 1057, pp. 3–17. Springer, Heidelberg (1996)
11. Casas-Garriga, G.: Discovering unbounded episodes in sequential data. In: Lavrač, N., Gamberger, D., Todorovski, L., Blockeel, H. (eds.) PKDD 2003. LNCS (LNAI), vol. 2838, pp. 83–94. Springer, Heidelberg (2003)
12. Ding, B., Lo, D., Han, J., Khoo, S.-C.: Efficient mining of closed repetitive gapped subsequences from a sequence database. In: ICDE 2009: Proc. of the 25th Int. Conf. on Data Engineering. IEEE Computer Society, Los Alamitos (2009)